Age-Length Key

Alex J. Benecke

January 31, 2017

Age-length keys will be produced from the measured TL and the age-at-capture of each Largemouth Bass. All samples will be divided into 25 mm bins? size bins (denoted L_i) and grouped by age (A_j). The probability that a fish is a particular age given its size ($p_{j|i}$) will be calculated by dividing the number of fish (n_{ji}) in the i^th length interval of the j^th age by the total number of fish in that size interval. The calculation for $p_{j|i}$ will be performed in R using the FSA, magrittr, and dplyr packages according to the methods described by Derek Ogle (2016a) (Bache and Wickham 2016, Ogle 2016b, Wickham and Francois 2016).

```
library(FSA)
```

```
## ## FSA v0.8.17. See citation('FSA') if used in publication.
## ## Run fishR() for related website and fishR('IFAR') for related book.
```

```
library(magrittr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(nnet)
```

```
LMB <- read.csv("Data/Clean-Data/2016_largemouth-bass_clean.csv") %>% select(FID,
    Site, AgeCap, LenCap, WTg, SEXCON, Sex)

LMB$FID <- factor(LMB$FID)
LMB$Site <- factor(LMB$Site)
LMB$SEXCON <- factor(LMB$SEXCON)
LMB$Sex <- factor(LMB$Sex)

str(LMB)
```

```
## 'data.frame':    131 obs. of  7 variables:
##  $ FID   : Factor w/ 131 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ Site  : Factor w/ 11 levels "2","4","6","8",..: 6 6 6 7 10 10 10 10 10 10 ...
##  $ AgeCap: int  4 4 4 6 4 NA 1 2 4 8 ...
##  $ LenCap: int  347 292 348 374 375 355 195 289 388 423 ...
##  $ WTg   : int  658 415 557 669 716 719 118 479 986 1258 ...
##  $ SEXCON: Factor w/ 5 levels "0","1","3","6",..: 5 3 3 5 3 5 5 3 3 5 ...
##  $ Sex   : Factor w/ 3 levels "0","1","2": 3 2 2 3 2 3 3 2 2 3 ...
```

```
headtail(LMB)
```

```
##     FID  Site AgeCap LenCap WTg SEXCON Sex
## 1     1    11      4    347 658      8   2
## 2     2    11      4    292 415      3   1
## 3     3    11      4    348 557      3   1
## 129 130    15      2    266 305      8   2
## 130 131    15      2    261 282      3   1
## 131 132 15972      7    395 971      3   1
```

```
LMB %<>% mutate(lencat25 = lencat(LenCap, w = 25))
LMB %<>% mutate(lencat20 = lencat(LenCap, w = 20))
headtail(LMB)
```

```
##     FID  Site AgeCap LenCap WTg SEXCON Sex lencat25 lencat20
## 1     1    11      4    347 658      8   2      325      340
## 2     2    11      4    292 415      3   1      275      280
## 3     3    11      4    348 557      3   1      325      340
## 129 130    15      2    266 305      8   2      250      260
## 130 131    15      2    261 282      3   1      250      260
## 131 132 15972      7    395 971      3   1      375      380
```

```
is.na(LMB$AgeCap)
```

```
##   [1] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
##  [12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [23] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
##  [34] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [45] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [56] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [67] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [78]  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [89] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [100] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [111] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [122] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
LMB.unaged <- filter(LMB, is.na(AgeCap))
headtail(LMB.unaged)
```

```
##    FID Site AgeCap LenCap WTg SEXCON Sex lencat25 lencat20
## 1    6   18     NA    355 719      8   2      350      340
## 2   28   18     NA    193 125      6   2      175      180
## 3   79    8     NA    169  71      6   2      150      160
## 31  79    8     NA    169  71      6   2      150      160
## 4   81    8     NA    166  74      1   1      150      160
## 5  124   15     NA    202 136      1   1      200      200
```

```
all(is.na(LMB.unaged$AgeCap))  # Better be True
```

```
## [1] TRUE
```

```
LMB.aged <- filter(LMB, !is.na(AgeCap))
headtail(LMB.aged)
```

```
##     FID  Site AgeCap LenCap WTg SEXCON Sex lencat25 lencat20
## 1     1    11      4    347 658      8   2      325      340
## 2     2    11      4    292 415      3   1      275      280
## 3     3    11      4    348 557      3   1      325      340
## 124 130    15      2    266 305      8   2      250      260
## 125 131    15      2    261 282      3   1      250      260
## 126 132 15972      7    395 971      3   1      375      380
```

```r
any(is.na(LMB.aged$AgeCap))  # Better be False
```

```
## [1] FALSE
```

```r
(alk.freq <- xtabs(~lencat20 + AgeCap, data = LMB.aged))
```

```
##         AgeCap
## lencat20  1  2  3  4  5  6  7  8
##      100  1  0  0  0  0  0  0  0
##      120  5  0  0  0  0  0  0  0
##      140  4  0  0  0  0  0  0  0
##      160  5  0  0  0  0  0  0  0
##      180  6  0  0  0  0  0  0  0
##      200  3  1  0  0  0  0  0  0
##      220  6  2  0  0  0  0  0  0
##      240  0 10  0  0  0  0  0  0
##      260  0 15  0  0  0  0  0  0
##      280  0 12  1  1  0  0  0  0
##      300  0  7  5  0  0  0  0  0
##      320  0  1  1  4  1  0  0  0
##      340  0  0  1  5  5  0  0  0
##      360  0  0  0 10  1  1  1  0
##      380  0  0  0  3  2  0  2  0
##      400  0  0  0  1  1  0  0  0
##      420  0  0  0  0  0  1  0  1
```

```r
rowSums(alk.freq)
```

```
## 100 120 140 160 180 200 220 240 260 280 300 320 340 360 380 400 420 
##   1   5   4   5   6   4   8  10  15  14  12   7  11  13   7   2   2
```

```r
alk <- prop.table(alk.freq, margin = 1)
round(alk, 3)
```

```
##         AgeCap
## lencat20     1     2     3     4     5     6     7     8
##      100 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      120 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      140 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      160 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      180 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      200 0.750 0.250 0.000 0.000 0.000 0.000 0.000 0.000
##      220 0.750 0.250 0.000 0.000 0.000 0.000 0.000 0.000
##      240 0.000 1.000 0.000 0.000 0.000 0.000 0.000 0.000
##      260 0.000 1.000 0.000 0.000 0.000 0.000 0.000 0.000
##      280 0.000 0.857 0.071 0.071 0.000 0.000 0.000 0.000
##      300 0.000 0.583 0.417 0.000 0.000 0.000 0.000 0.000
##      320 0.000 0.143 0.143 0.571 0.143 0.000 0.000 0.000
##      340 0.000 0.000 0.091 0.455 0.455 0.000 0.000 0.000
##      360 0.000 0.000 0.000 0.769 0.077 0.077 0.077 0.000
##      380 0.000 0.000 0.000 0.429 0.286 0.000 0.286 0.000
##      400 0.000 0.000 0.000 0.500 0.500 0.000 0.000 0.000
##      420 0.000 0.000 0.000 0.000 0.000 0.500 0.000 0.500
```

```
### Some weirdness here I have a 450 mm 2 year old (FID 55 removed) and 425 mm
### 6 yr old while my 8 year old is 400 mm ? Check ages on some of these!!!
```

```
LMB.mlr <- multinom(AgeCap ~ lencat20, data = LMB.aged, maxit = 500)
```

```
## # weights:  24 (14 variable)
## initial  value 262.009634
## iter  10 value 165.708694
## iter  20 value 86.522821
## iter  30 value 76.911878
## iter  40 value 76.530563
## iter  50 value 75.522690
## iter  60 value 74.878320
## iter  70 value 74.842334
## iter  80 value 74.810713
## iter  90 value 74.779561
## iter 100 value 74.772647
## iter 110 value 74.757476
## iter 120 value 74.705796
## iter 130 value 74.683333
## iter 140 value 74.674652
## iter 150 value 74.671280
## final  value 74.671273
## converged
```

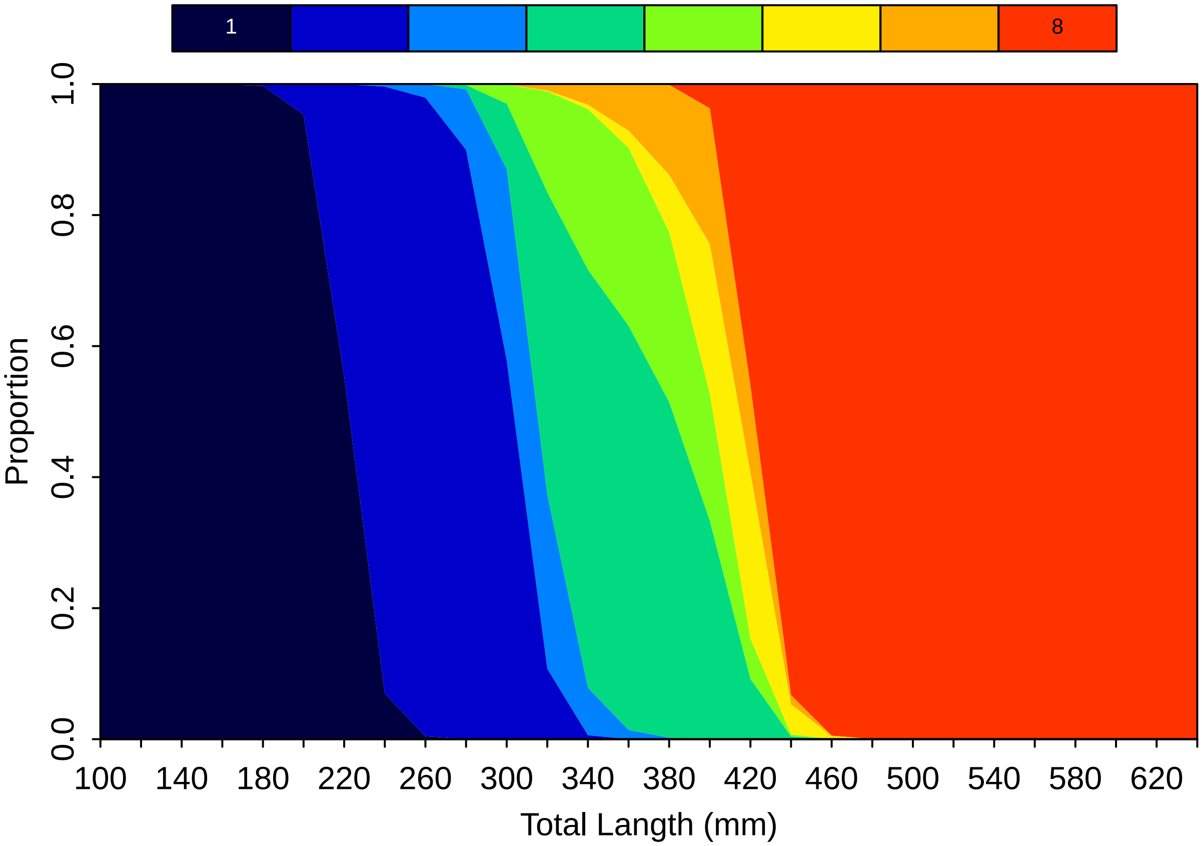
```
lens <- seq(100, 650, 20)
alk.sm <- predict(LMB.mlr, data.frame(lencat20 = lens), type = "probs")
row.names(alk.sm) <- lens  # for clarity
round(alk.sm, 3)  #for display purposes
```

```
##         1     2     3     4     5     6     7     8
## 100 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## 120 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## 140 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## 160 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## 180 0.997 0.003 0.000 0.000 0.000 0.000 0.000 0.000
## 200 0.954 0.046 0.000 0.000 0.000 0.000 0.000 0.000
## 220 0.554 0.446 0.000 0.000 0.000 0.000 0.000 0.000
## 240 0.070 0.926 0.004 0.000 0.000 0.000 0.000 0.000
## 260 0.004 0.975 0.020 0.000 0.000 0.000 0.000 0.000
## 280 0.000 0.899 0.093 0.006 0.002 0.000 0.000 0.000
## 300 0.000 0.578 0.292 0.100 0.029 0.000 0.001 0.000
## 320 0.000 0.107 0.266 0.462 0.154 0.001 0.010 0.000
## 340 0.000 0.006 0.072 0.638 0.245 0.007 0.032 0.000
## 360 0.000 0.000 0.014 0.617 0.272 0.026 0.071 0.000
## 380 0.000 0.000 0.002 0.512 0.259 0.088 0.137 0.001
## 400 0.000 0.000 0.000 0.333 0.194 0.229 0.207 0.037
## 420 0.000 0.000 0.000 0.092 0.061 0.253 0.133 0.460
## 440 0.000 0.000 0.000 0.004 0.003 0.046 0.014 0.933
## 460 0.000 0.000 0.000 0.000 0.000 0.004 0.001 0.995
## 480 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 500 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 520 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 540 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 560 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 580 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 600 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 620 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
## 640 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
```

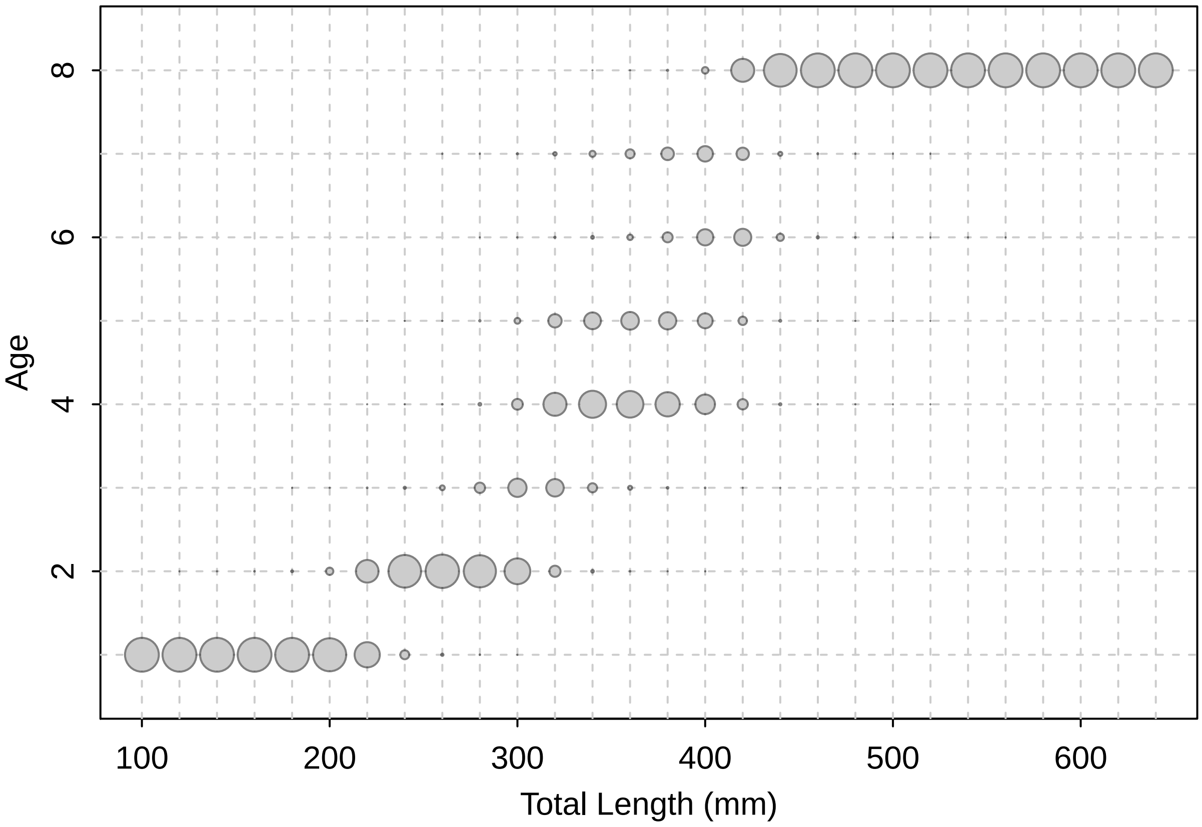
```
### Much better but still double check the ages from before!!!
```

Smoothed Age-Length Key Model

```
### Smoothed ALK Model
alkPlot(alk.sm, type = "area", showLegend = TRUE, leg.cex = 0.7, xlab = "Total Langth (mm)")
```

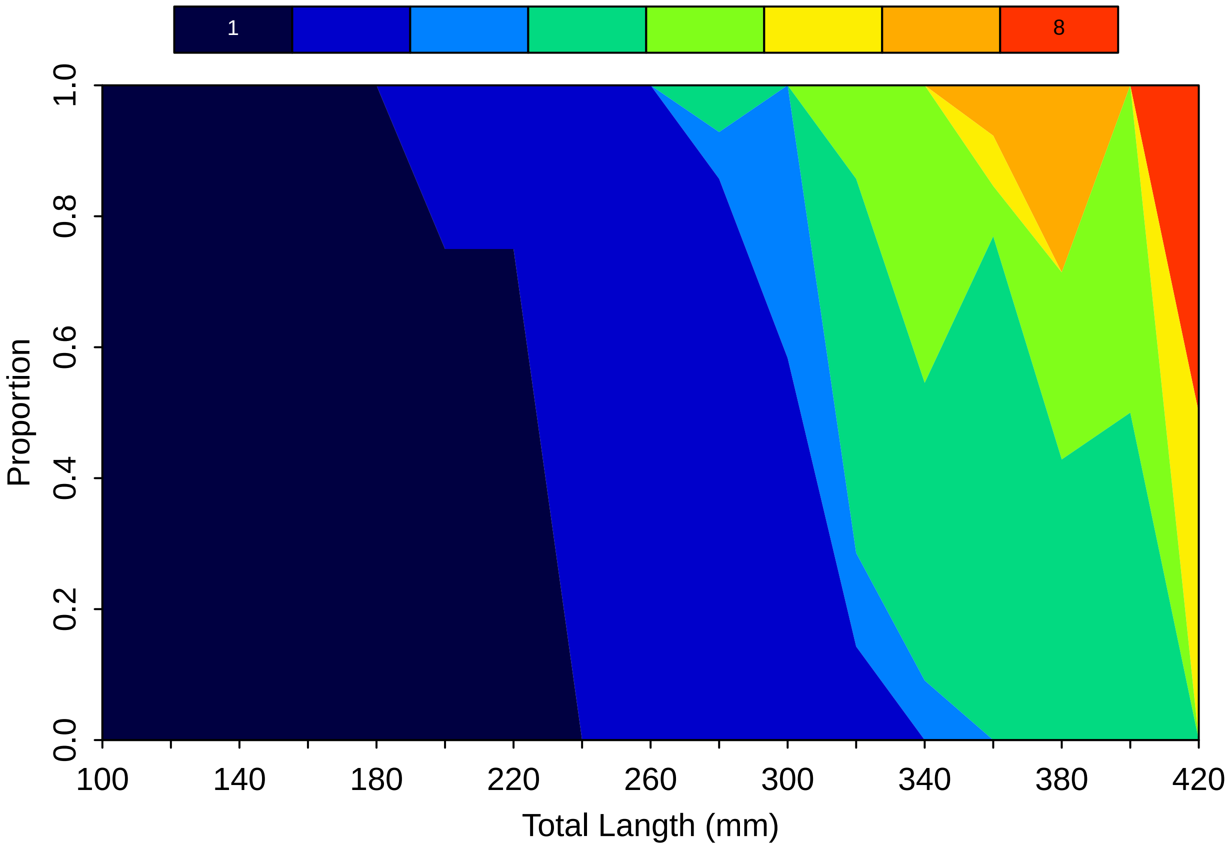


```
alkPlot(alk.sm, type = "bubble", xlab = "Total Length (mm)")
```

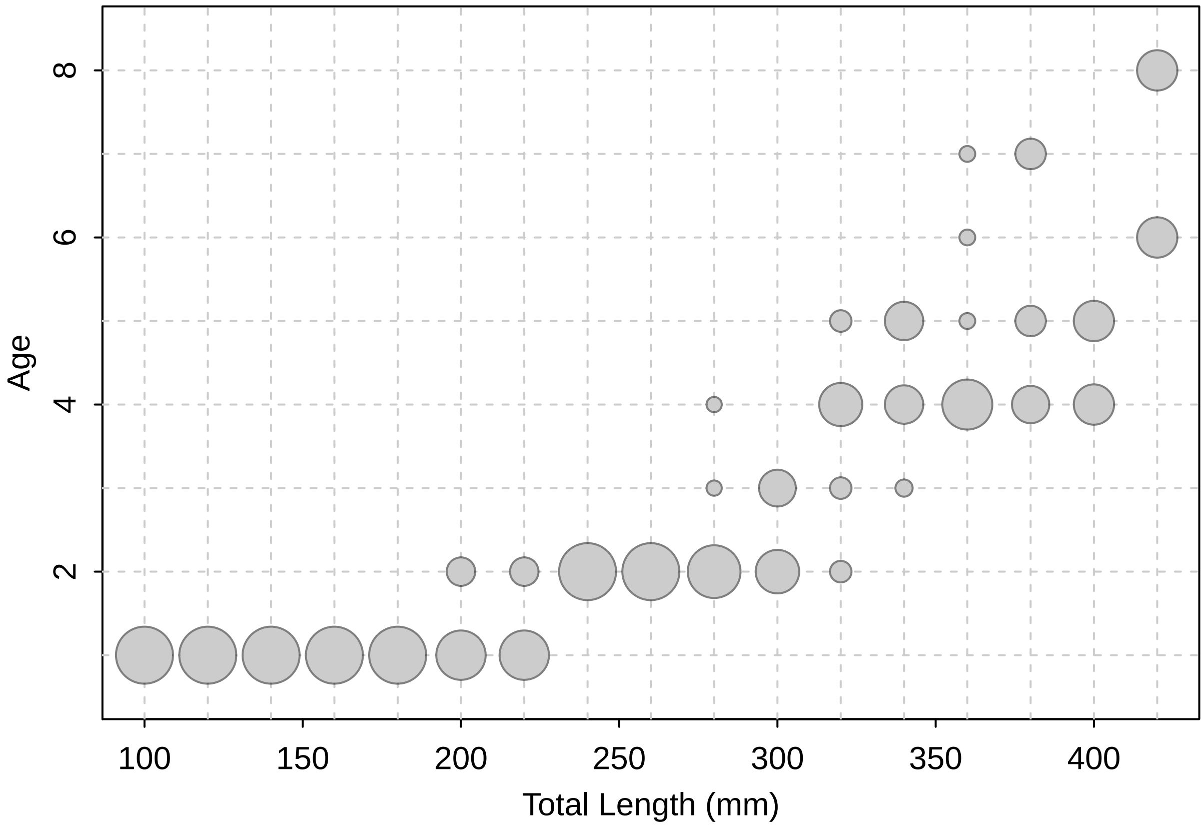


Age-Length Key

```
alkPlot(alk, type = "area", showLegend = TRUE, leg.cex = 0.7, xlab = "Total Langth (mm)")
```



```r
alkPlot(alk, type = "bubble", xlab = "Total Length (mm)")
```



```
### Doesn't Look Good!!!
```